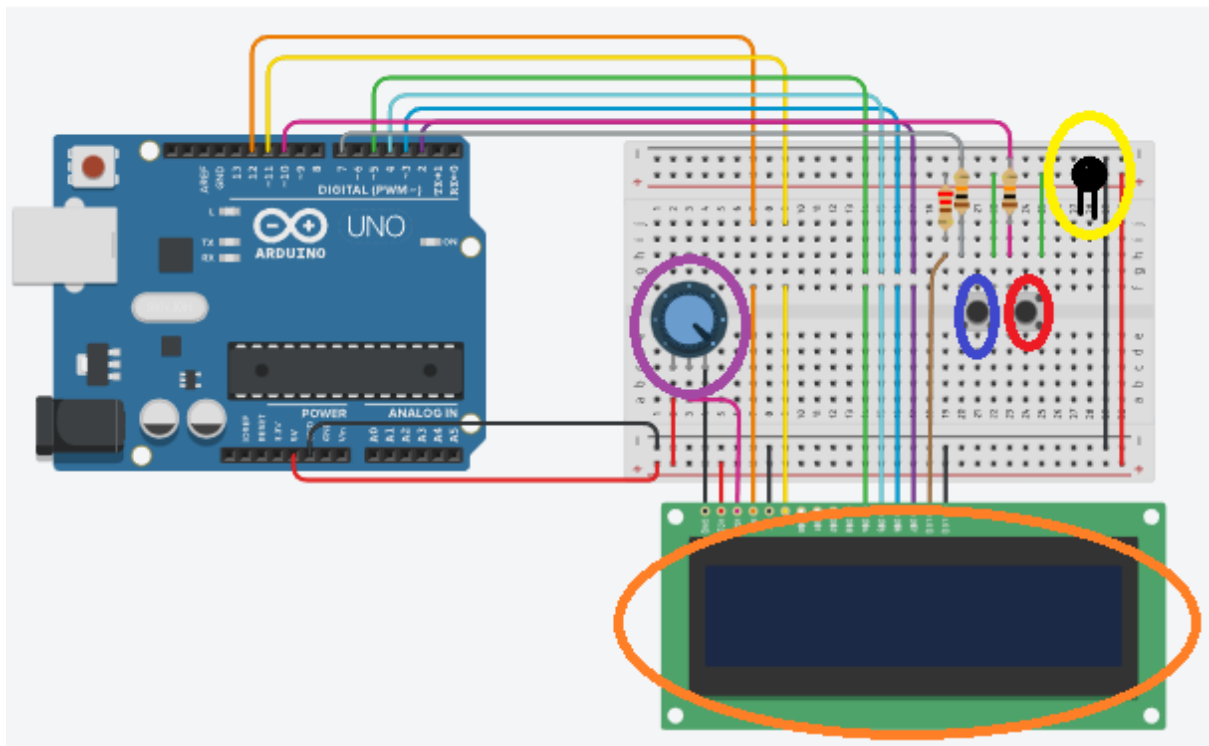


Código del final de labo 1

Este código implementa una funcionalidad similar a la famosa “ranita” de los Simpson, pero en código Morse.



Utiliza dos botones y bucles condicionales infinitos para procesar la entrada y generar una salida en una pantalla LCD. A continuación, detallamos los componentes utilizados y su funcionalidad:



Botón rojo: Permite al usuario ingresar texto en código Morse.

Botón azul: Actúa como la tecla “Enter” y agrega espacios entre las letras en Morse.

Pantallita naranja: Muestra el texto ingresado en código Morse.

Potenciómetro violeta: Controla el brillo de la pantalla (aunque no es crucial para la funcionalidad principal).

Brazzer amarillo: Genera un sonido (aproximado) correspondiente al código Morse ingresado.

Explicación del código:

El código está diseñado para funcionar en un Arduino con botones que permiten ingresar texto en código Morse y mostrarlo en una **Pantallita** LCD. También utiliza un **Brazzer** para generar el sonido correspondiente al código morse ingresado.

1) Se incluye la biblioteca LiquidCrystal, que se utiliza para controlar la **pantalla LCD**. Luego se declaran algunas variables y se instancia un objeto "lcd" de la clase LiquidCrystal para establecer la configuración de la **Pantallita**.

2) En la función **setup()**, se inicia la comunicación serial a una velocidad de 9600 baudios, se inicia la **Pantallita** LCD y se configuran los pines de los **botones** como entradas con resistencias de pull-up internas habilitadas.

3) En la función **loop()**, se lee el estado del **botón** de inicio utilizando **digitalRead()**. Si el botón está en estado HIGH (presionado), se llama a la función **cuanToTiempo()** para determinar la duración del botón presionado. Si el tiempo es inferior a 0.05 segundos, se asigna un valor de 1(punto) al elemento correspondiente del vector. Si el tiempo es superior a 0.5 segundos, se asigna un valor de 2(barritas). Después, se incrementa la variable **i** para seguir un orden en la entrada de los símbolos morse.

4) Se lee el estado del **botón** de "finalización". Si el botón está en estado HIGH (presionado), se llama a la función **EscribirLetra()** para determinar qué letra en código morse se ingresó (LA PARTE DE LOS IFs ETERNOS) y mostrarla en la **Pantallita** LCD. Luego se restablecen los valores del vector y la variable **i**.

5) La función **cuanToTiempo()** se utiliza para medir la duración del **botón** de inicio presionado. Se mide el tiempo en milisegundos utilizando la función **millis()**. La función espera hasta que el **botón** de inicio se suelte (estado LOW) y luego calcula la diferencia entre el tiempo actual y el tiempo inicial (delta tiempo), el tiempo se divide por 1000 para convertirlo a segundos y se imprime a través de la comunicación serial. El valor de tiempo se devuelve como resultado de la función.

6) La función ***EscribirLetra()*** se utiliza para determinar qué letra en código morse se ingresó según los valores almacenados en el vector. Dependiendo de la combinación de valores en el vector, se muestra la letra correspondiente en la **Pantallita** LCD. Cada letra se muestra durante 600 milisegundos antes de continuar con la siguiente. Se utiliza la función ***delay()*** para generar la pausa.

7) Por último, el **Brazzer** recibe la letra y chilla para que se escuche.